

# IMAGE PROCESSING APPARATUS AND METHOD FOR DISPLAYING PICTURE-IN-PICTURE WITH FRAME RATE CONVERSION

## BACKGROUND OF THE INVENTION

### 1. Field of the Invention

The present invention relates to an image processing system, and more particularly, to an image processing apparatus and a method for implementing picture-in-picture (PIP) with frame rate conversion.

### 2. Description of the Related Art

In a conventional television (TV), one channel is displayed on a display device. However, a plurality of channels can be displayed on the display device of a TV using a feature called picture-in-picture (PIP). PIP is an image processing method for simultaneously displaying another channel on part of the display device. In prior systems, in order to display PIP, two frame memories for storing asynchronously input image data have been used. Since the size of an image processing apparatus for displaying the PIP is increased by frame memories, such systems can be too large.

In a case where frame rates of input signals are different from that of the display device, a frame rate conversion device is used to adjust the frame rates of the input signals. A conventional frame rate conversion device prevents data from being compromised by synchronizing a clock used as a frame buffer clock with the frequency of each of the input signals using a phase locked loop (PLL). However, in case of using a PLL, the size of a circuit is increased, and a method for operating a frame buffer becomes complicated. Thus, this method is not effective.

Also, for the above-mentioned PIP and frame rate conversion, there are the following problems. For example, assuming that there are two input signals which are asynchronous data, when two input signals are displayed on one display device, the two input signals must be synchronized with each other. Further, in a case where there are limitations in that the display device can not perform a multi-sync function for generating various synchronized signals owing to physical and technical characteristics, a function for simultaneously converting the two input signals to an

output frame rate of the display device must be provided. For example, in the case of a liquid crystal device (LCD) monitor, the output frame rate of a display signal in a SXGA level (1280 x 1024) monitor is physically restricted to 75KHz or less. Also, the output frame rate in a UXGA level (1600 x 1200) monitor is restricted to 60Hz or less. That is, in a case where input signals above the output frame rate are displayed on the display device, the frame rates of the input signals must be reduced. Also, in a case where the frame rates of the two input signals are different, and there is one display device, the frame rates of the two input signals must be independently converted to the output frame rate of the display device. However, in a case where PIP and frame rate conversion are simultaneously performed, a process of synchronizing the two input signals is complicated. During the process, data may be damaged, and the size of a system may be increased by an increase in frame memories.

#### SUMMARY OF THE INVENTION

To solve the above problems, it is a first objective of the present invention to provide an image processing apparatus capable of displaying picture-in-picture (PIP) using one frame buffer to effectively operate memory and capable of performing frame rate conversion without damaging data, even in a case where an input signal is not synchronized with an output signal, by using a simple data synchronizing circuit.

It is a second objective to provide an image processing method for displaying PIP with frame rate conversion implemented by the image processing apparatus.

In accordance with the invention, there is provided an image processing apparatus for displaying on a display device a plurality of input data asynchronously input through different channels and converting frame rates of each of the input data in accordance with an output frame rate of the display device. The image processing apparatus includes an input buffer unit, a data synchronizing unit, first, second, and third memories, and a memory control unit. The input buffer unit buffers input data, which are externally and asynchronously input through two or more channels, using different input clock signals and outputs buffered data as first data and first data enabling signals. The data synchronizing unit synchronizes the

first data output from the input buffer unit with an output clock signal in response to one of the different input clock signals and the first data enabling signals and outputs synchronized data as second data and second data enabling signals in response to each of the first data enabling signals. The first memory multiplexes the second data according to time sharing, stores the second data in different regions, and outputs stored data in response to a first memory enabling signal. The second memory writes and reads data output from the first memory in response to a frame buffer control signal. The third memory stores data output from the second memory and outputs the stored data as a display signal in response to a second memory enabling signal. The memory control unit generates the first memory enabling signal to control data flow between the first memory and the second memory, generates the frame buffer control signal to control frame rates of the first and second data and the display signal, and generates the second memory enabling signal to control data flow between the second memory and the third memory.

In accordance with the invention, there is also provided an image processing method for displaying on one display device a plurality of input data asynchronously input through different channels and converting frame rates of the input data in accordance with an output frame rate of the display device. The method comprises the steps of (a) buffering the plurality of input data using each of input clock signals and synchronizing each of buffered data with an output clock signal, (b) storing the plurality of input data synchronized with the output clock signal in a first memory in response to input enabling signals, (c) comparing a write address with a read address of the first memory to determine whether data stored in the first memory are stored in a second memory, (d) comparing the frame rates of the plurality of input data with that of an output display signal to control data write and read of the second memory, and (e) comparing a write address with a read address of a third memory to determine whether output data of the second memory are also stored in the second memory, and outputting data stored in the third memory as a display signal for displaying on the display device.

#### BRIEF DESCRIPTION OF THE DRAWINGS

The foregoing and other objects, features and advantages of the invention

will be apparent from the more particular description of a preferred embodiment of the invention, as illustrated in the accompanying drawings in which like reference characters refer to the same parts throughout the different views. The drawings are not necessarily to scale, emphasis instead being placed upon illustrating the principles of the invention.

FIG. 1 is a block diagram of an image processing apparatus for implementing picture-in-picture (PIP) with frame rate conversion according to an embodiment of the present invention.

FIG. 2 is a block diagram of a first data synchronizing unit of the image processing apparatus shown in FIG. 1.

FIGS. 3A through 3J are timing diagrams illustrating operation of a frequency conversion module of the first data synchronizing unit shown in FIG. 2.

FIG. 4 is a block diagram of a second data synchronizing unit of the image processing apparatus shown in FIG. 1.

FIGS. 5A and 5B are flow charts illustrating an image processing method for implementing PIP with frame rate conversion according to the embodiment of the present invention.

FIG. 6 illustrates input/output data flow of a first first-in first-out (FIFO) of the image processing apparatus shown in FIG. 1.

FIGS. 7A and 7B illustrate operation of a frame buffer of the image processing apparatus shown in FIG. 1.

FIG. 8 illustrates input/output data flow of a second FIFO of the image processing apparatus shown in FIG. 1.

#### DETAILED DESCRIPTION OF THE INVENTION

FIG. 1 is a block diagram of an image processing apparatus for implementing picture-in-picture (PIP) with frame rate conversion according to an embodiment of the present invention. Referring FIG. 1, the image processing apparatus includes a first input buffer 100, a second input buffer 105, a first data synchronizing unit 110, a second synchronizing unit 115, a first first-in first-out (FIFO) 120, a frame buffer 130, a second FIFO 140, a memory control unit 150, and a color space converting unit 160.

The first input buffer 100 buffers a first input data IN1 input through a first channel in response to a first clock signal and outputs buffered data as a first data and a first data enabling signal. Here, assuming that the first input data IN1 is graphic data that is externally input, for example, R, G, and B signals, then the first clock signal is a graphic clock signal CK\_G for buffering the graphic data, and the first data and the first data enabling signal, which are output from the first input buffer 100, are a first graphic data GDATA1 and a first graphic enabling signal GEN1, respectively. Also, the first graphic data GDATA1 output from the first input buffer 100 is blocked in response to a first input blocking signal, that is, a graphic input blocking signal IN\_BLK\_G, output from the memory control unit 150 and is intercepted so that it is not input into the first data synchronizing unit 110 in a blocked state.

The second input buffer 105 buffers a second input data IN2 input through a second channel in response to a second clock signal and outputs buffered data as a second data and a second data enabling signal. Here, assuming that the second input data IN2 is video data that is externally input, for example, a luminance signal (Y) and a color-difference signal (U/V), then the second clock signal is a video clock signal CK\_V for buffering the video data, and the second data and the second data enabling signal, which are output from the second input buffer 105, are a first video data VDATA1 and a first video enabling signal VEN1, respectively. Also, the first video data VDATA1 output from the second input buffer 105 is blocked in response to a second input blocking signal, that is, a video input blocking signal IN\_BLK\_V, output from the memory control unit 150 and is intercepted so that it is not input into the second data synchronizing unit 110 in a blocked state.

In the embodiment of FIG. 1, a case of using two input buffers is illustrated. However, more than two input buffers may be used depending on the design of the image processing apparatus. Also, the first and second input buffers 100 and 105 may be merged into one input buffer unit and named accordingly.

The first graphic data GDATA1 is input into the first data synchronizing unit 110 in response to the graphic clock signal CK\_G and the first graphic enabling signal GEN1. The first data synchronizing unit 110 synchronizes the first graphic data GDATA1 with an output clock signal CK\_O. Also, the first data synchronizing

unit 110 outputs the first graphic data GDATA1 synchronized with the output clock signal CK\_O as a second graphic data GDATA2 and a second graphic enabling signal GEN2 in response to a delayed video enabling signal DVEN.

The first video data VDATA1 is input into the second data synchronizing unit 115 in response to the first video enabling signal VEN1 and the video clock signal CK\_V, which are output from the second input buffer 105. The second data synchronizing unit 115 synchronizes the first video data VDATA1 with the output clock signal CK\_O. Also, the second data synchronizing unit 115 outputs the first video data VDATA1 synchronized with the output clock signal CK\_O as a second video data VDATA2 and a second video enabling signal VEN2. Here, the second video data VDATA2 and the second video enabling signal VEN2 are output to the color space converting unit 160. The first and second data synchronizing units 110 and 115 may be merged into one data synchronizing unit and named accordingly.

The color space converting unit 160 converts the second video data VDATA2 output from the second data synchronizing unit 115 into a graphic data of R/G/B and outputs converted video data VGDATA. Also, the color space converting unit 160 delays the second video enabling signal VEN2 for a predetermined time and outputs the delayed video enabling signal DVEN to the memory control unit 150 and the first data synchronizing unit 110. The color space converting unit 160 is selectively used and may be not used depending on the design of a system. The converted video data VGDATA, which is the second video data VDATA2 converted into a graphic data in the color space converting unit 160, is applied to the first FIFO 120.

The memory control unit 150 generates a first FIFO enabling signal FEN1 to control data flow between the first FIFO 120 and the frame buffer 130 and generates a frame buffer control signal FBCON to control frame rates of data input into the frame buffer 130 and an output display signal. Also, the memory control unit 150 generates a second FIFO enabling signal FEN2 to control data flow between the frame buffer 130 and the second FIFO 140. For these operations, the memory control unit 150 includes a first FIFO control unit 152, a frame buffer control unit 154, and a second FIFO control unit 156.

Specifically, the first FIFO control unit 152 generates the first FIFO enabling signal FEN1 in response to the second graphic enabling signal GEN2 output from

the first data synchronizing unit 110, in response to a delayed first video enabling signal DVEN, and in response to a first frame data enabling signal FDEN1. Also, the first FIFO control unit 152 generates a write address and a read address of the first FIFO 120, detects underflow (UNF) between the write address and the read address, and controls data write and read of the first FIFO 120 according to a detected result. Here, the write address and the read address are generated by circular addressing. In addition, the first FIFO control unit 152 outputs the underflow UNF to the frame buffer control unit 154 and receives the first frame data enabling signal FDEN1 corresponding to the underflow UNF from the frame buffer control unit 154 to generate the first FIFO enabling signal FEN1. Here, the first frame data enabling signal FDEN1 is a signal for indicating a write interval and a read interval of the frame buffer 130 and for indicating a write interval of a graphic data and a write interval of a video data.

The frame buffer control unit 154 compares a frame rate of the data input into the frame buffer 130 with that of an output display signal, generates the graphic and video input blocking signals IN\_BLK\_G and IN\_BLK\_V for blocking data of the first and second input buffers 100 and 105 in response to the compared result, and outputs the graphic and video input blocking signals IN\_BLK\_G and IN\_BLK\_V to the first and second buffers 100 and 105. Also, the frame buffer control unit 154 generates the first and second frame data enabling signals FDEN1 and FDEN2 and the frame buffer control signal FBCON by using the underflow UNF detected from the first FIFO control unit 152 and by using overflow (OVF) detected from the second FIFO control unit 156. Frame rate conversion implemented in the frame buffer control unit 154 will be described in detail with reference to FIGS. 5A and 5B. Also, the frame buffer control unit 154 enables the second frame data enabling signal FDEN2 when data is read from the frame buffer 130 to the second FIFO 140.

The second FIFO control unit 156 generates the second FIFO enabling signal FEN2 in response to the second frame data enabling signal FDEN2 and an output enabling signal OUT\_EN. Also, the second FIFO control unit 156 generates a write address and a read address of the second FIFO 140, detects overflow OVF between the write address and read address of the second FIFO 140 to control data write and read of the second FIFO 140 according to a detected result. Here, the output

enabling signal OUT\_EN is a signal generated from a timing generating circuit (not shown), and data is output from the second FIFO 140 according to an output display format. For example, the output display format may be decided by considering information such as the number of horizontal lines and the number of vertical lines.

The first FIFO 120 includes different storing regions and exclusively stores the second graphic data GDATA2 output from the first data synchronizing unit 110 and the converted video data VGDATA output from the color space converting unit 160 in response to the first FIFO enabling signal FEN1 output from the first FIFO control unit 152 and the output clock signal CK\_O, in each of the different storing regions. Here, data output from the first FIFO 120 is referred to as first FIFO output data FDATA1.

The frame buffer 130 stores the first FIFO output data FDATA1 in response to the frame buffer control signal FBCON output from the frame buffer control unit 154 and the output clock signal CK\_O and outputs a stored data FBDATA. Here, data output from the frame buffer 130 is referred to as frame buffer output data FBDATA.

The frame buffer 130 may be realized as memory for storing at least one frame of data of the input video data.

The frame buffer output data FBDATA input into the second FIFO 140, and the second FIFO 140 outputs data in response to the second FIFO enabling signal FEN2 output from the second FIFO control unit 156. Here, the data output from the second FIFO 140 is referred to as second FIFO output data FDATA2 and is output as a display signal through an output terminal OUT.

Referring to FIG. 1, signals left of the dotted line are asynchronous, and signals right of the dotted line are synchronized with the output clock signal CK\_O.

FIG. 2 is a block diagram of the first data synchronizing unit 110 of the image processing apparatus shown in FIG. 1. Referring to FIG. 2, the first data synchronizing unit 110 includes a write address counter 200, a demultiplexer 210, a parallel buffer 220, a multiplexer 230, a read address counter 240, a frequency conversion module 250, an underflow detecting unit 260, and a graphic enabling signal generating unit 270.

The write address counter 200 counts a write address of the parallel buffer 220 in response to a graphic clock signal CK\_G and a first graphic enabling signal

GEN1 and outputs a counted write address CNT\_WADD. The demultiplexer 210 demultiplexes input data in response to the counted write address CNT\_WADD output from the write address counter 200 and selectively outputs data demultiplexed by the demultiplexer 210 to one of a number (N) of parallel registers 220\_1, . . . , 220\_n of the parallel buffer 220.

The parallel buffer 220 includes a number of (N) parallel registers 220\_1, . . . , 220\_n and stores data D1, . . . , Dn demultiplexed in the demultiplexer 210. As shown in FIG. 2, the first graphic data GDATA1 is stored in one of the parallel registers 220\_1, . . . , 220\_n in response to the graphic clock signal CK\_G.

The frequency conversion module 250 converts the frequency region of the counted write address CNT\_WADD into the frequency of an output clock signal CK\_O and outputs a converted frequency write address FC\_WADD. Operation of the frequency conversion module 250 will be described in detail with reference to FIG. 3.

The read address counter 240 counts a read address of the parallel buffer 220 in response to the output clock signal CK\_O and a second graphic enabling signal GEN2 and outputs a counted read address CNT\_RADD.

The underflow detecting unit 260 receives the counted read address CNT\_RADD output from the read address counter 240 and the converted frequency write address FC\_WADD, thereby generating an underflow UND of the parallel buffer 220.

The graphic enabling signal generating unit 270 generates the second graphic enabling signal GEN2 in response to the underflow UND. Preferably, an AND operation is performed on the underflow UND and a delayed video enabling signal DVEN, thereby generating the second graphic enabling signal GEN2. For this operation, the graphic enabling signal generating unit 270 includes inverters 272 and 276, and an AND gate 274. The inverter 272 inverts the underflow UND detected in the underflow detecting unit 260 and outputs an inverted underflow signal. The inverter 276 inverts the delayed video enabling signal DVEN and outputs an inverted delayed video enabling signal. The AND gate 274 performs an AND operation on an output signal of the inverter 272 and an output signal of the inverter 276 and outputs the ANDed signal as the second graphic enabling signal GEN2.

The multiplexer 230 receives data output from one of the registers 220\_1, . . . , 220\_n of the parallel buffer 220 and selectively outputs the data as a second graphic data GDATA2 in response to the counted read address CNT\_RADD output from the read address counter 240.

Hereinafter, operation of the first data synchronizing unit 110 shown in FIG. 2 will be described in greater detail. First, when the first graphic enabling signal GEN1 is applied through the first input buffer 100, the write address counter 200 sequentially counts a write address of the parallel buffer 220 in response to the graphic clock signal CK\_G. Here, data output from the demultiplexer 210 in response to the counted write address CNT\_WADD are stored in one of the parallel registers 220\_1, . . . , 220\_n of the parallel buffer 220. Here, the frequency of the counted write address CNT\_WADD is converted and is generated as FC\_WADD, and the underflow UND is detected in response to the converted frequency write address CNT\_WADD and the counted read address CNT\_RADD. That is, in a case where there is no further data to be output from the parallel buffer 220 due to a slow increase in the write address while the read address is increasing, the underflow UND is set to a predetermined level, for example, to a high level. In this case, the second graphic enabling signal GEN2 generated from the graphic enabling signal generating unit 270 is at a low level. Thus, data output from the first data synchronizing unit 110 to the first FIFO 120 through the multiplexer 230 is regarded as invalid data. However, in a case where the underflow UND is not detected and is set as a low level and then the delayed second video enabling signal DVEN is at a low level, the second graphic enabling signal GEN2 is enabled at a high level. Thus, the read address counter 240 is enabled, and a graphic data output to the first FIFO 120 through the multiplexer 230 is regarded as valid data.

FIGS. 3A through 3J are timing diagrams illustrating operation of a frequency conversion module 250 of the first data synchronizing unit 110 shown in FIG. 2. FIG. 3A denotes an input clock signal, for example, a graphic clock signal CK\_G, FIG. 3B denotes a parallel buffer write address WADD, FIG. 3C denotes an even write address WADD\_E, and FIG. 3D denotes an odd write address WADD\_O. Also, FIG. 3E denotes a selection signal SEL\_I generated by the graphic clock signal CK\_G, FIG. 3F denotes an output clock signal CK\_O, FIG. 3G denotes a

synchronized even write address WADD\_EO synchronized with the output clock signal CK\_O, FIG. 3H denotes a synchronized odd write address WADD\_OO synchronized with the output clock signal CK\_O, FIG. 3I denotes a synchronized selection signal SEL\_IO synchronized with the output clock signal CK\_O, and FIG. 3J denotes the converted frequency write address FC\_WADD.

Operation of the frequency conversion module 250 of FIG. 2 will be described with reference to FIG. 3 as follows. The parallel buffer write address WADD is synchronized with the graphic clock signal CK\_G shown in FIG. 2 and is divided into an even write address WADD\_E shown in FIG. 3C and an odd write address WADD\_O shown in FIG. 3D. When the output clock signal CK\_O is input, as shown in FIG. 3F, the selection signal SEL\_I shown in FIG. 3E is synchronized with the output clock signal CK\_O, resulting in the synchronized selection signal SEL\_IO of FIG. 3I. Here, the even write address WADD\_E and the odd write address WADD\_O are synchronized with the output clock signal CK\_O, resulting in the synchronized even write address WADD\_EO shown in FIG. 3G and the synchronized odd write address WADD\_OO of FIG. 3H, respectively.

For example, as indicated by times V1 through V3, when a timing violation occurs, each of the even and odd synchronized write addresses WADD\_EO and WADD\_OO and the selection signal SEL\_IO can be inaccurately marked as intervals **a** through **c**. Here, an assumption is made that the synchronized odd write address WADD\_OO is output when the synchronized selection signal SEL\_IO of FIG. 3I is at a low level, and the even write address WADD\_EO is output when the selection signal SEL\_IO is at a high level. That is, even though each of the intervals **a** and **b** is inaccurate due to the time V1 and the time V2, the synchronized selection signal SEL\_IO is at a low level for the interval **a**, and the SEL\_IO is at a high level for the interval **b**, thus the converted frequency write address FC\_WADD cannot be normally output. Also, at the time V3, it does not matter whether an address D of FIG. 3J output at the interval **c** of the synchronized selection signal SEL\_IO of FIG. 3I is an even number or an odd number. That is, the converted frequency write address FC\_WADD may be address 3 or address 4.

FIG. 4 is a block diagram of a portion of the second data synchronizing unit 115 of FIG. 1. The second data synchronizing unit 115 has a similar structure to

that of the first data synchronizing unit 110, except for a difference in the structure of an output terminal. Thus, description and illustration of the same elements will be omitted, and only an underflow detecting unit 30 and a video enabling signal generating unit 32 are shown in FIG. 4.

Referring to FIG. 4, the video enabling signal generating unit 32 is comprised of an inverter 34 for inverting an underflow UND detected from the underflow detecting unit 30. That is, the inverter 34 inverts the underflow UND detected from the underflow detecting unit 30 of FIG. 4 and outputs the result as a second video enabling signal VEN2. Although not specifically shown, the second video enabling signal VEN2 is input into a read address counter (not shown), and simultaneously, the VEN2 is output as a delayed video enabling signal DVEN through the color space converting unit 160 of FIG. 1.

In this way, in a state where the second video enabling signal VEN2 is enabled in the first and second data synchronizing units 110 and 115 shown in FIGS. 2 through 4, a video data VDATA2 is input into the first FIFO 120. Here, the second graphic enabling signal GEN2 is disabled. Thus, the second graphic data GDATA2 is not input to the first FIFO 120 and denotes a state of being stored in the parallel buffer 220. On the other hand, when the second graphic enabling signal GEN2 is enabled, the graphic data stored in the parallel buffer 220 is input to the first FIFO 120. Here, since the second video enabling signal VEN2 is disabled, the video data is not input to the first FIFO 120, but instead stored in the parallel buffer 220.

FIGS. 5A and 5B are flow charts illustrating an image processing method for implementing PIP with frame rate conversion according to an embodiment of the present invention. Operation of the image processing apparatus according to the present invention and method thereof will be described in detail with reference to FIGS. 1 through 5.

First, referring to FIG. 5A, first and second input data IN1 and IN2, which are input by each of the input clock signals, that is, the graphic clock signal CK\_G and the video clock signal CK\_V, through different channels for PIP, are buffered, and each of the buffered data is synchronized with the output clock signal CK\_O in step 500. That is, step 500 is performed in the first and second input buffers 100 and

105 and the first and second data synchronizing units 110 and 115 of FIG. 1, and the first and second input data IN1 and IN2 are a graphic data and a video data, respectively. Referring to FIG. 1, the buffered data, that is, a first graphic data GDATA1 and a first video data VDATA1, are applied to the first and second data synchronizing units 110 and 115 with the first graphic and first video enabling signals GEN1 and VEN1. That is, the first graphic data GDATA1 and the first video data VDATA1 synchronized with each of the input clock signals CK\_G and CK\_V are synchronized with the output clock signal CK\_O in the first and second data synchronizing units 110 and 115 and are output as the second graphic data GDATA2 and the second video data VDATA2, respectively, as shown in FIGS. 2 through 4.

After step 500, the second graphic data GDATA2 and the second video data VDATA2 synchronized with the output clock signal are stored in the first FIFO 120 in response to each of the input enabling signals, that is, a second graphic enabling signal GEN2 and a second video enabling signal VEN2 in step 510. More specifically, first, it is determined whether the second graphic enabling signal GEN2 for the first input data, that is, the second graphic data GDATA2, is enabled in step 512. If the second graphic enabling signal GEN2 is enabled, the second graphic data GDATA2 synchronized with the output clock signal CK\_O is stored in the first FIFO 120 in step 514. On the other hand, if the second graphic enabling signal GEN2 is disabled in step 512, then, it is determined whether the second video enabling signal VEN2 is enabled in step 516. If the second video enabling signal VEN2 is enabled in the step 516, the second video data VDATA2 synchronized with the output clock signal CK\_O is stored in the first FIFO 120 in step 518. Here, data write and read are performed in the first FIFO 120.

Then, it is determined whether data of the first FIFO 120 are stored in the frame buffer 130 by comparing the write address of the first FIFO 120 with the read address of the first FIFO 120 in step 520. Specifically, it is determined whether an underflow UNF occurs between the write address and the read address of the first FIFO 120 in step 522. In step 522, the underflow UNF is obtained as follows. For example, in a case where the write address of the first FIFO 120 is larger than the

read address of the first FIFO 120, if the following condition is satisfied, it is regarded that the underflow UNF occurs.

[Equation 1]

$$WADD\_F1 - RADD\_F1 < TH1$$

Here, WADD\_F1 denotes a first FIFO write address, RADD\_F1 denotes a first FIFO read address, and TH1 is a threshold value which may be arbitrarily set by a user. In a case where the difference between the first FIFO write address WADD\_F1 and the first FIFO read address RADD\_F1 is smaller than the threshold value TH1, that is, in a state where the first FIFO write address WADD\_F1 is larger than the first FIFO read address RADD\_F1, it is determined that underflow UNF occurs in the first FIFO control unit 152 and the underflow UNF is set to a high level.

That is, if it is determined that underflow UNF occurs, data of the first FIFO 120 are not output to the frame buffer 130 in step 526. Specifically, the frame buffer control unit 154 receives the underflow UNF from the first FIFO control unit 152, outputs a first frame enabling signal FDEN1 to the first FIFO control unit 152, and stops outputting data from the first FIFO 120. However, if the difference between the first FIFO write address WADD\_F1 and the first FIFO read address RADD\_F1 is larger than the threshold value TH1, underflow UNF does not occur and is maintained at a low level. That is, if it is judged that the underflow UNF does not occur, the data of the first FIFO 120 are stored in the frame buffer 130 in step 524. Here, it is determined by the first frame data enabling signal FDEN1 output from the frame buffer control unit 154 whether it is the graphic data or the video data, which are stored in the first FIFO 120, that is stored.

Also, in a case where the first FIFO write address WADD\_F1 is not larger than the first FIFO read address RADD\_F1, it is determined that the underflow UNF is detected even in a case where the following condition is satisfied.

[Equation 2]

$$N1 + WADD\_F1 - RADD\_F1 < TH1$$

Here, N1 denotes a first FIFO address size. In other words, in case of the graphic data, N1 denotes the address size of a graphic data region in the first FIFO

120, and in case of the video data, N1 denotes the address of a video data region in the first FIFO 120. That is, referring to Equation 2, if a value in which the first FIFO address size N1 is added to the difference of the first FIFO write address WADD\_F1 and the first FIFO read address RADD\_F1 is smaller than the threshold value TH1, it is determined that the underflow UNF occurs. Here, step 526 of FIG. 5 is performed. In another case, it is determined that underflow UNF does not occur, step 524 of FIG. 5 is performed. Likewise, the underflow UNF of the Equations 1 and 2 occurring in step 520 can be commonly applied to the graphic data and the video data. Data flow between the first FIFO 120 and the frame buffer 130 performed in step 520 will be described in detail with reference to FIG. 6.

Referring to FIG. 6, VG\_DELIMIT denotes a storing address delimiter for dividing an address region into the video data region and the graphic data region. As described previously, since the graphic data and the video data are exclusively input into the first FIFO 120, regions for storing the graphic data and the video data are divided. Thus, each write address for the graphic data and the video data is independently generated. Reference numeral 62 of FIG. 6 denotes video data input into the first FIFO 120, and reference numeral 64 denotes video data output from the first FIFO 120. Also, reference numeral 66 denotes graphic data input into the first FIFO 120, and reference numeral 68 denotes graphic data output from the first FIFO 120. That is, in order to perform a frame rate conversion adequate for a display device while implementing PIP, the frequency of the output clock signal CK\_O must be determined so that addition of the rate of the graphic data 66 synchronized with the graphic clock signal CK\_G and the rate of the video data 62 input by being synchronized with the video clock signal CK\_V may be accepted.

An interval  $T_p$  denotes a frame buffer data read/write interval, an interval  $T_w$  denotes a data write interval where data are output from the first FIFO 120 to the frame buffer 130, and  $T_R$  denotes a data read interval where data are output from the frame buffer 130 to the second FIFO 140.  $T_G$  and  $T_V$  denote a graphic data write interval and a video data write interval, respectively, and are exclusively used by time sharing in the first FIFO 120. The size of the first FIFO 120 is determined by the intervals of  $T_w$  and  $T_R$ , and in the first FIFO 120, the storing address delimiter

VG\_DELIMIT is determined by rates of the video data input 62 and the graphic data input 66 input into the first FIFO 120.

Referring to FIG. 6, UNF1 illustrates a case where underflow is detected in the graphic data region, and UNF2 illustrates a case where the underflow is detected in the video data region. For example, when the graphic data stored in the first FIFO 120 is read from the frame buffer 130, if the rate of output data is faster than that of input data, the underflow UNF, preferably, graphic data region underflow UNF1, is output from the first FIFO control unit 152 to the frame buffer control unit 154. Thus, the frame buffer control unit 154 is notified so that the frame buffer 130 does not request additional data. As described previously, the first frame data enabling signal FDEN1 includes an interval setting signal for indicating the data write interval  $T_w$  and data read interval  $T_R$ , and an interval setting signal for indicating the graphic data write interval  $T_G$  and video data write interval  $T_V$ . Thus, when underflow UNF occurs, the interval setting signal for indicating the data write interval  $T_w$  is disabled, and thus, additional data are not output from the first FIFO 120 to the frame buffer 130. For a case where the underflow UNF is video region underflow UNF2, the same operations are performed.

Referring back to FIG. 5A, in step 520, when data is input/output from the first FIFO 120 to the frame buffer 130, data write and read of the frame buffer 130 are controlled by the result in which the frame rate of the stored graphic data or video data is compared with that of the output display signal in step 530. Specifically, it is judged whether the frame rate of the graphic data or video data is faster than the output frame data of the output display signal in step 532. Here, if it is judged that the frame rate of the graphic data or video data is faster than that of the display signal, the frame buffer control unit 154 outputs graphic and video input blocking signals IN\_BLK\_G and IN\_BLK\_V to the first and second input buffers 100 and 105. Thus, data is not input into the first and second data synchronizing units 110 and 115 by controlling the first and second input buffers 100 and 105 and by blocking one frame of the graphic data or video data in step 534. For example, it is determined whether the graphic input blocking signal IN\_BLK\_G or the video input blocking signal IN\_BLK\_V generated from the frame buffer control unit 154 occurs according to the following condition.

[Equation 3]

$$M - F + P_i + R_{fr} > F$$

Here, M denotes a frame buffer storage size, F is the image size of one frame,  $P_i$  is an interval of a frame of data being read as one frame data is input into the frame buffer 130, and  $R_{fr}$  is an output frame rate/input frame rate. Thus, if the condition of Equation 3 is satisfied, the graphic input blocking signal IN\_BLK\_G or the video input blocking signal IN\_BLK\_V is disabled and is maintained at a low level. That the graphic input blocking signal IN\_BLK\_G is at a low level means a normal state. That is, even though a next frame data is written in the frame buffer 130 at a rapid speed while an arbitrary frame data stored in the frame buffer 130 is being read, the next frame data is led by the read data, thereby preventing data from being damaged. However, if the condition of Equation 3 is not satisfied, the graphic input blocking signal IN\_BLK\_G is enabled at a high level, preventing data from being output from the first input buffer 100 of FIG. 1 to the first data synchronizing unit 110. Thus, data are not input into the frame buffer 130 through the first data synchronizing unit 110 and the first FIFO 120. Thus, damage to data caused by the frame rate of the frame buffer 130 can be prevented.

FIGS. 7A and 7B illustrate operation of a frame buffer of the apparatus shown in FIG. 1. Referring to FIG. 7A, an interval from a start time of the present read frame to a read time when the next input frame begins is indicated as  $P_i$ . Also, an interval from a start time of the next input frame to a start time of the present read frame is indicated as M-F. Additionally, an interval from a read time when the next input frame begins to a read time expected when the next input frame ends is indicated as  $R_{fr}$ .

Referring back to FIG. 5A, if it is judged that the frame rate of the graphic or video data is not faster than an output frame rate of the display signal in step 532, it is determined whether the frame rate of the graphic or video data is slower than the output frame rate of the display signal in step 536. If the frame rate of the graphic or video data is not slower than the output frame rate of the display signal in step 536, then step 524 is performed. However, if it is judged that the frame rate of the graphic or video data is slower than the output frame rate of the display signal, one frame of the graphic or video data stored in the frame buffer 130 is repeated and

read in step 538. That is, it is determined whether the frame data is repeated and read is determined by the following conditions.

[Equation 4]

$$P_i + \frac{1}{R - f} > F, \text{ and}$$

$$P_i > M - F$$

That is, in a case where the conditions of Equation 4 is satisfied, a signal for repeating and reading the frame data is disabled. However, in a case where the conditions of Equation 4 is not satisfied, the frame buffer 130 repeats one frame of the output data previously output by the frame buffer control signal FBCON and then outputs the output data. Here, assuming that a signal for repeating and reading the data of the frame buffer 130 is RE\_read and is included in the frame buffer control signal FBCON, the signal RE\_read is at a low level when the condition of Equation 4 is satisfied. Thus, data corresponding to the next address of the frame data currently output from the frame buffer 130 are consecutively output. However, if the RE\_read is at a high level, the data previously output from the frame buffer 130 is reread. Reread operation of the data stored in the frame buffer 130 will be described with reference to FIG. 7B.

Referring to FIG. 7B, an interval from a start time of the previous write frame to a write time when the next read frame begins is indicated as M-F, and an interval from a start time of the previous write frame to a read time expected when the next read frame ends is indicated as 1/R\_fr (that is, input frame rate/output frame rate). Also, an interval from a start time F-1 of the present write frame to a write time when the next read frame begins is indicated as P<sub>i</sub>.

In the present invention, the frame rate for the input and/or output data can be raised or lowered by performing step 530. Thus, in the image processing apparatus and method thereof according to the present invention, a user can programmably implement a desired frame rate of the display device during frame rate conversion.

Therefore, regardless of the characteristics of input data and limitations in the output display device, the input data can be stably displayed on the display device.

Referring to FIG. 5B, after step 538, data output from the frame buffer 130 are stored in the second FIFO 140. That is, it is determined whether the data are stored from the frame buffer 130 to the second FIFO 140 by comparing the write address with the read address of the second FIFO 140, and stored data are output as a display signal in step 540. Specifically, it is judged whether the overflow OVF occurs between the write address and the read address of the second FIFO 140 in step 542. Here, overflow OVF indicates that the data input into the second FIFO 140 are lead by output data, and it can be determined whether overflow OVF occurs by the address of the second FIFO 140. Here, occurrence of overflow OVF is determined according to the satisfaction of the following condition in a case where the write address of the second FIFO 140 is larger than the read address of the second FIFO 140.

[Equation 5]

$$N2 + RADD\_F2 - WADD\_F2 < TH2$$

Here, N2 is a second FIFO address size, WADD\_F2 is a second FIFO write address, RADD\_F2 is a second FIFO 140 read address, and TH2 is a threshold value which is set by a user. Thus, if a value of the second FIFO write address WADD\_F2 subtracted from the sum of the second FIFO read address RADD\_F2 and the address size N2 of the second FIFO 140 is smaller than the threshold value TH2, it is determined that overflow OVF occurs, and here, the overflow OVF is at a high level. However, if the conditions of Equation 5 is not satisfied, it is determined that the overflow OVF does not occur, and the OVF is at a low level. Also, in a case where the second FIFO write address WADD\_F2 is not larger than the second FIFO read address RADD\_F2, if the following condition is satisfied, it is determined that overflow OVF occurs.

[Equation 6]

$$RADD\_F2 - WADD\_F2 < TH2$$

That is, in a case where the difference between the second FIFO read address RADD\_F2 and the second FIFO write address WADD\_F2 is smaller than the threshold value TH2, it is determined that overflow OVF occurs. However, if the condition of Equation 6 is not satisfied, overflow OVF does not occur and is maintained at a low level. When referring to Equations 5 and 6, if it is determined that overflow OVF occurs in step 542, data stored in the frame buffer 130 are not output to the second FIFO 140 in step 548. However, if it is judged that overflow OVF does not occur in step 542, data of the frame buffer 130 are stored in the second FIFO 140 in step 544. Also, the data stored in the second FIFO 140 are output as a display signal in accordance with an output display format in step 546.

FIG. 8 illustrates data input/output flow of the second FIFO 140 of the image processing apparatus shown in FIG. 1. Referring to FIG. 8, reference numeral 82 denotes data input into the second FIFO 140, and reference numeral 84 denotes data output of the second FIFO 140. That is, data read into the second FIFO 140 is performed only at a data read interval  $T_R$  of the frame buffer 130, and data is not input to the second FIFO 140 at other intervals. The frame buffer control unit 154 enables the second frame data enabling signal FDEN2 at the data read interval  $T_R$  where the data is read from the frame buffer 130 to the second FIFO 140 so that the data may be input to the second FIFO 140. However, the second frame data enabling signal FDEN2 applied from the frame buffer control unit 154 to the second FIFO control unit 156 is disabled at an interval where overflow OVF occurs between the two data read intervals  $T_R$ , and thus, data is not input from the frame buffer 130 to the second FIFO 140.

Frame rate conversion for the display device while implementing PIP can be performed according to the above-mentioned steps.

According to the present invention, memories such as first and second FIFOs 120 and 140 and the frame buffer 130 can be effectively operated, and efficiency in the system size can be increased by using only one frame buffer when implementing PIP. Also, frame rate conversion is performed by using a simple data synchronizing circuit without using the PLL, thereby preventing damage to data and performing frame rate conversion using a minimum amount of memory. Also, the image processing apparatus according to the present invention can programmably

implement a desired frame rate of the display device during frame rate conversion, thereby supporting various input formats.

While this invention has been particularly shown and described with references to preferred embodiments thereof, it will be understood by those skilled in the art that various changes in form and details may be made therein without departing from the spirit and scope of the invention as defined by the appended claims.

2025.03.04 10:44